



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/488,976	01/21/2000	William J. Baer	STL000014US1	5177
23373	7590	06/20/2006	EXAMINER	
SUGHRUE MION, PLLC 2100 PENNSYLVANIA AVENUE, N.W. SUITE 800 WASHINGTON, DC 20037			PHAM, HUNG Q	
			ART UNIT	PAPER NUMBER
			2168	

DATE MAILED: 06/20/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	<b>Application No.</b>	<b>Applicant(s)</b>	
	09/488,976	BAER ET AL.	
	<b>Examiner</b>	<b>Art Unit</b>	
	HUNG Q. PHAM	2168	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) Responsive to communication(s) filed on 21 March 2006.
- 2a) This action is **FINAL**.                            2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) Claim(s) 1-32 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) Claim(s) \_\_\_\_\_ is/are allowed.
- 6) Claim(s) 1-32 is/are rejected.
- 7) Claim(s) \_\_\_\_\_ is/are objected to.
- 8) Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on \_\_\_\_\_ is/are: a) accepted or b) objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) All    b) Some \* c) None of:
  1. Certified copies of the priority documents have been received.
  2. Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) <input type="checkbox"/> Notice of References Cited (PTO-892)	4) <input type="checkbox"/> Interview Summary (PTO-413)
2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)	Paper No(s)/Mail Date. _____ .
3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____ .	5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)
	6) <input type="checkbox"/> Other: _____ .

## DETAILED ACTION

### ***Response to Arguments***

#### ***Rejection under 35 U.S.C. § 112, first paragraph***

Applicants' arguments with respect to the rejection of claim 31 under 35 U.S.C. § 112, first paragraph, have been fully considered. The rejection of claim 31 under 35 U.S.C. § 112, first paragraph, has been withdrawn in view of the amendment of claim 31.

#### ***Objection***

Applicants' arguments with respect to the objection of claims 1, 4, 9, 12, 17 and 20 have been fully considered. The objection of claims 1, 4, 9, 12, 17 and 20 has been withdrawn in view of the amendment of claims 1, 4, 9, 12, 17 and 20.

#### ***Rejection under 35 U.S.C. § 102(b) and 103(a)***

Applicants' arguments with respect to the rejection of claims 1-30 under 35 U.S.C. § 102(b) and 102(a) has been considered but they are not persuasive.

As argued by applicants:

(a) From page 13, line 17 to page 14, line 4:

*...In particular, the identifier ("Introduction") of the user-provided content cannot be added to the list of content entity identifiers (outline as shown on left hand side of page 7) as the Examiner asserts since the user-provided content ("Introduction") has already been stored in the data repository in the "receiving" step (outline stored on the C drive according to the Examiner). Therefore, the Examiner is citing the same operation of Vaughn, the addition of content to the outline, for teaching distinguishable aspects of the claim.*

*Since the Examiner has not shown that all of the elements of claims are disclosed in the cited art, claim 1 and its dependent claims should be deemed allowable.*

(b) At page 14, lines 3-6:

*Since claims 4, 9, 12, 17, 20 and 31 recite similar elements, claims 4, 9, 12, 17, 20 and 31 and their dependent claims should be deemed allowable for at least the same reasons.*

(c) At page 14, lines 8-15:

*To the extent claim 31 recites elements similar to claim 1, it should be deemed allowable for at least the same reasons set forth above.*

*Claim 31 recites "storing said custom content object in said one or more object servers; and storing information specifying the custom content object and attribute information concerning the custom content object in the library server." The Examiner asserts that ezWriter folder "1ezWriter" discloses the claimed object server. However, the ezWriter folder does not teach or suggest the claimed object server which would be apparent to one of ordinary skill in the art.*

Examiner respectfully disagrees.

(a) Vaughn discloses all the elements of claimed invention.

As taught by Vaughn (Page 7), to create a collection of RTF files represented by an ezWriter file that is an index, a user selects [File, New], types in the name that the user would like to call, e.g., "The Trifanton Castle Trilogy" then selects <OK> (FIG. 1). An ezWriter file, and "The Trifanton Castle Trilogy" file are created (FIG. 2). The outline, which is an index file, has only one entry and is displayed in the Outline Window (FIG. 3). Both files were placed into a directory with the same name under the ezWriter directory (FIG. 4). As seen, a collection of RTF files under "The Trifanton Castle Trilogy" file, as *the content object is defined by an outline as a list of content entity identifiers*.

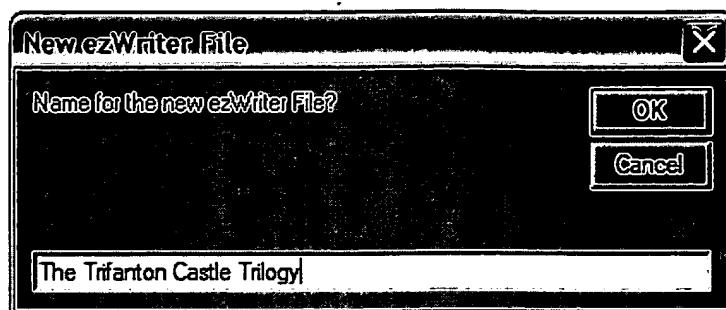


FIG. 1

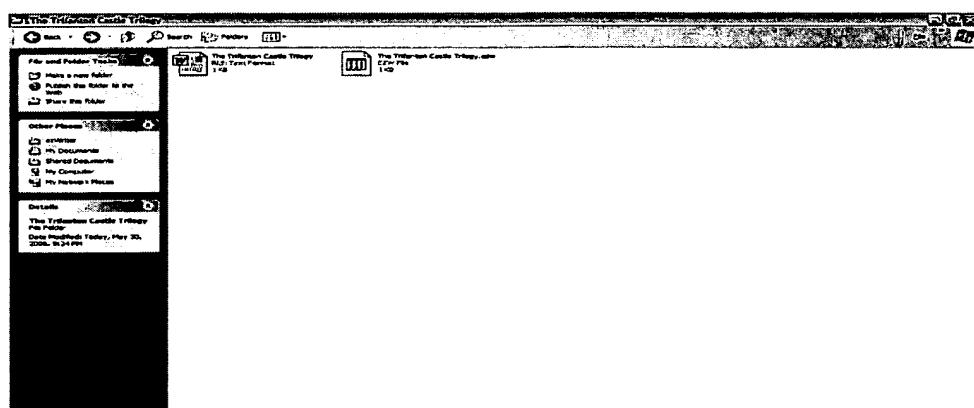


FIG. 2



FIG. 3

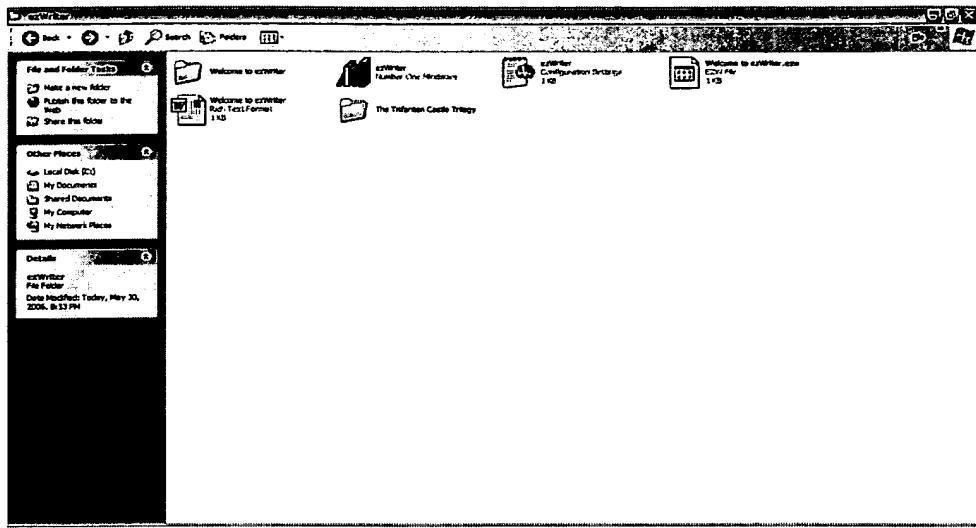


FIG. 4

As taught by Vaughn, ezWriter works in Windows 95, Windows 98, or Windows NT (Page 3, "What is ezWriter"). A user can create an RTF file using Microsoft Word 97, save it and send it to some one. RTF documents can include texts and pictures (Page 6, "Welcome to ezWriter"). Thus, an RTF document contains user-provided texts, e.g., "This is the content of the First Chapter", is equivalent to the claimed limitation *user-provided content*, which is received by the system from the user that has ezWriter, must be *assigned* a file name, e.g., "First Chapter", as *an identifier* when saving to "The Trifanton Castle Trilogy" directory. By saving, *user-provided content* (RTF file) with *its identifier* (file name) are *stored in the data repository*, e.g., "The Trifanton Castle Trilogy" directory.

As further disclosed by Vaughn (Page 8, "How to Add Entries to the Outline"), the file name as *identifier of the user-provided content* could be *added to* the outline as *the list of content entity identifiers* by selecting the [Outline, Edit] of the Outline Window pull-down menu to view the .ezw file. The .ezw file is opened in Windows Notepad for adding the desired entry, e.g., "First Chapter" (FIG. 8). After saving the file and closing Notepad, by refreshing the outline [Outline, Refresh], the Outline Window displays the added file name (FIG. 9). By selecting "First Chapter"

in the Outline Window, *the user-provided content* (RTF file contains texts), is added to "The Trifanton Castle Trilogy" and displayed in the Writer Window (FIG. 10). In short, the technique as discussed indicates the claimed limitation *adding the identifier of the user-provided content to the list, whereby the user-provided content is added to the content object, wherein the user-provided content is content created by the user.*

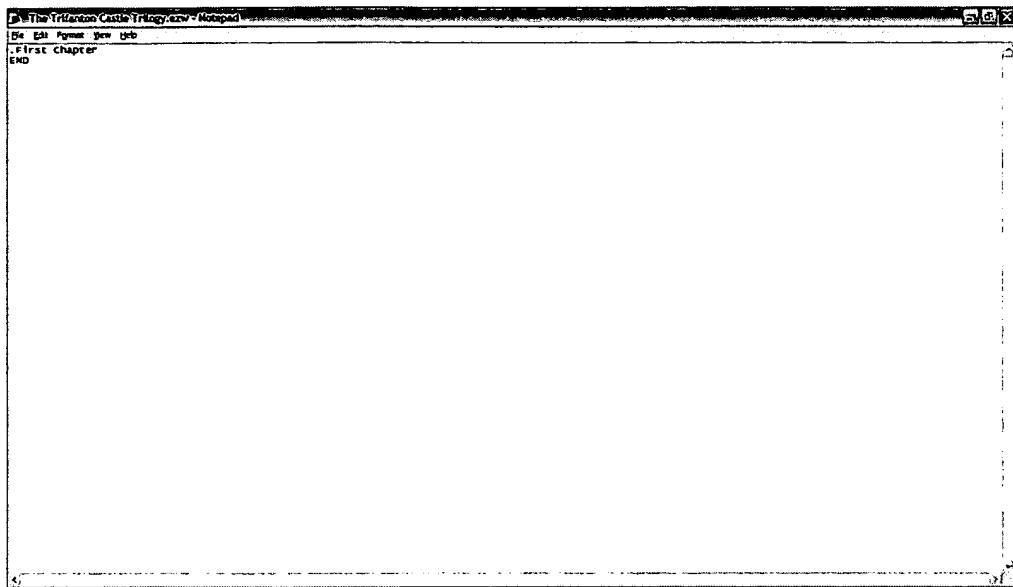


FIG. 8

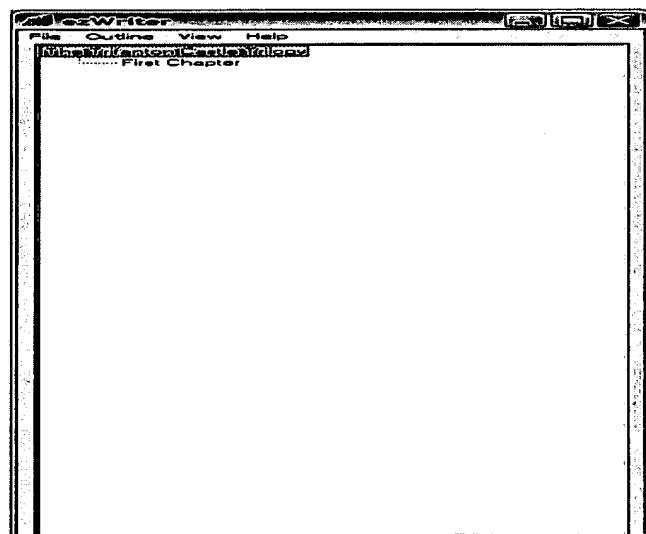


FIG. 9

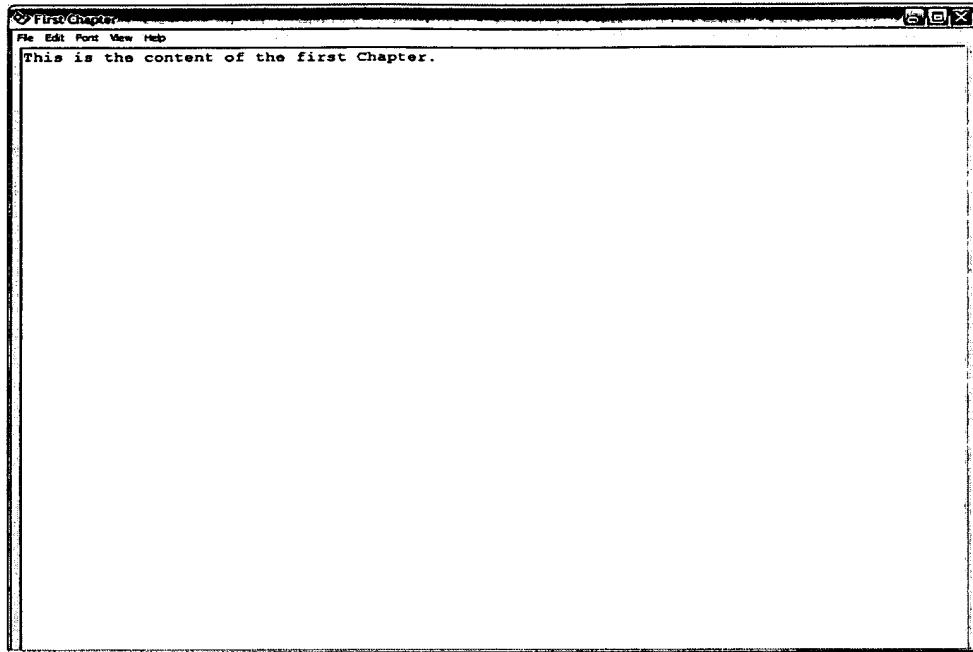


FIG. 10

(b) Dependent claims of claim 1 are unpatentable because of the reasons as discussed above. Claims 4, 9, 12, 17, 20 and 31 and their dependent claims recite similar elements, and are unpatentable because of the same reasons.

(c) A server is a computer delivers information to other computers linked by network, or a computer program that provides services to other computer programs in the same or other computers. The claimed limitation *object server* is considered as a computer that stores information. And this interpretation would be apparent to one of ordinary skill in the art.

In light of the foregoing arguments, the rejection under 35 U.S.C. § 102(b) and 103(a) is hereby sustained.

***Duplicate Claims, Warning***

Applicant is advised that should claims 4, 12 and 20 be found allowable, claims 1, 9 and 17 will be objected to under 37 CFR 1.75 as being a substantial duplicate thereof. When two claims in an application are duplicates or else are so close in content that they both cover the same thing, despite a slight difference in wording, it is proper after allowing one claim to object to the other as being a substantial duplicate of the allowed claim. See MPEP § 706.03(k).

***Claim Rejections - 35 USC § 102***

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

**Claims 1-31 are rejected under 35 U.S.C. 102(b) as anticipated by or, in the alternative, under 35 U.S.C. 103(a) as obvious over Vaughn [ezWriter 2.0 for Window].**

Regarding claims 1, 4, 9, 12, 17 and 20, Vaughn teaches *a computer-implemented method for adding user-provided content to a content object stored as a plurality of content entities in a data repository* (Page 2 and Page 15) comprising the steps of:

*having a user defining the content object by a list of content entity identifiers* (As taught by Vaughn (Page 7), to create a collection of RTF files represented by an ezWriter file that is an index, a user selects [File, New], types in the name that the user would like to call, e.g., "The Trifanton Castle Trilogy" then selects <OK> (FIG. 1). An ezWriter file, and "The Trifanton Castle Trilogy" file are created (FIG. 2). The outline, which is an index file, has only one entry and is displayed

in the Outline Window (FIG. 3). Both files were placed into a directory with the same name under the ezWriter directory (FIG. 4). As seen, a collection of RTF files under “The Trifanton Castle Trilogy” file, as *the content object is defined by an outline as a list of content entity identifiers* as in claims 1, 9, 17 or *a hierarchical outline of containers* as in claims 4, 12, 20);

*receiving user-provided content, assigning it an identifier, and storing it with its identifier in the data repository* (As taught by Vaughn, ezWriter works in Windows 95, Windows 98, or Windows NT (Page 3, “What is ezWriter”). A user can create an RTF file using Microsoft Word 97, save it and send it to some one. RTF documents can include texts and pictures (Page 6, “Welcome to ezWriter”). Thus, an RTF document contains user-provided texts, e.g., “This is the content of the First Chapter”, is equivalent to the claimed limitation *user-provided content*, which is received by the system from the user that has ezWriter, must be *assigned* a file name, e.g., “First Chapter”, as *an identifier* when saving to “The Trifanton Castle Trilogy” directory. By saving, *user-provided content* (RTF file) with *its identifier* (file name) are *stored in the data repository*, e.g., “The Trifanton Castle Trilogy” directory); and

*adding the identifier of the user-provided content to the list, whereby the user-provided content is added to the content object, wherein the user-provided content is content supplied or created by the user* (As further disclosed by Vaughn (Page 8, “How to Add Entries to the Outline”), the file name as *identifier of the user-provided content* could be *added to the outline as the list of content entity identifiers* by selecting the [Outline, Edit] of the Outline Window pull-down menu to view the .ezw file. The .ezw file is opened in Windows Notepad for adding the desired entry, e.g., “First Chapter” (FIG. 8). After saving the file and closing Notepad, by refreshing the outline [Outline, Refresh], the Outline Window displays the added file name (FIG. 9). By selecting “First Chapter” in the Outline Window, *the user-provided content* (RTF file contains texts) is added to “The Trifanton Castle Trilogy” and displayed in the Writer Window (FIG. 10). In short, the technique as discussed

indicates the claimed limitation *adding the identifier of the user-provided content to the list, whereby the user-provided content is added to the content object, wherein the user-provided content is content created by the user*).

Regarding claims 1, 4, 9, 12, 17 and 20, Vaughn teaches *a computer-implemented method for adding user-provided content to a content object stored as a plurality of content entities in a data repository* (*user-provided content*, e.g., the content as shown on the right hand side of Page 2, is added to *content object stored as a plurality of content entities in a data repository*, e.g., “Welcome to ezWriter” as shown in the screenshot of Page 15) comprising the steps of:

*having a user defining the content object by a list of content entity identifiers* (Page 7, The ezWriter File, paragraph 3, and the left hand side of Page 7, “Welcome to ezWriter” is considered as *content object*);

*adding the identifier of the user-provided content to the list* (Page 8, How to Add Entries to the Outline), *whereby the user-provided content is added to the content object* (Page 8, each added identifier will have an RTF file created for it, e.g., when adding “Introduction” to the list, an RTF file with file name “Introduction” as *user-provided content* is created and added to “Welcome to ezWriter”);

*receiving user-provided content* (Page 8, RTF file is received by the system when an entry is added to the outline, e.g., “Introduction” file), *assigning it an identifier* (Page 8, “Introduction” is assigned to the RTF file), *and storing it with its identifier in the data repository* (Page 8 and Page 15, the RTF file and its file name are stored in a directory in C drive as *the data repository*);

*wherein the user-provided content is content supplied or created by the user* (e.g., the content as shown on the right hand side of Page 2).

The difference between Vaughn and the claimed invention is the order of the process. In Vaughn teaching, the step of adding the identifier is processed before the step of receiving user-provided content, assigning an identifier to the content and storing the content and identifier.

However, as taught by Vaughn, ezWriter works in Windows 95, Windows 98, or Windows NT (Page 3, "What is ezWriter"). A user can create an RTF file using Microsoft Word 97, save it, and send it to someone. RTF documents can include texts and pictures (Page 6, "Welcome to ezWriter"). Thus, instead of creating an RTF file via ezWriter, the user can create an RTF file using Microsoft Word 97. When creating, the RTF files as *user-provided content* is received by the system, the file will be assigned by the user a file name as *an identifier*, and stored in the same directory of ezWriter File in C drive as *the data repository*. After creating the RTF file, the file name will be added to the outline as discussed above as disclosed by Vaughn at Page 9. When the user selects an entry in the outline, the corresponding RTF file has file name matches with the entry will be displayed in the Writer Window (these steps were tested by examiner, and ezWriter gave no error).

Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the Vaughn technique by using a conventional word processor to create an RTF file instead of using ezWriter, and by doing this, user-provided content could be created by a user at one site and transferring to another site that has ezWriter for compiling.

Regarding claims 2, 10 and 18, Vaughn teaches all the claim subject matters as discussed above with respect to claims 1, 9 and 17, Vaughn reference further comprising the step of *receiving a user-provided location for inserting the identifier of the user-provided content into the content object, and inserting the identifier into the list at that location* (the screenshot of Page 9).

Regarding claims 3, 11 and 19, Vaughn teaches all the claim subject matters as discussed above with respect to claims 2, 10 and 18, Vaughn reference further comprising the steps of *providing a user interface communicating with the data repository, and providing mechanisms for receiving the user-provided content and specification of a desired location through the user interface* (Pages 9 and 10, How the ezWriter File Works).

Regarding claims 5, 13 and 21, Vaughn teaches all the claim subject matters as discussed above with respect to claims 4, 12 and 20, the Vaughn reference further comprising the step of *receiving a user-provided location for inserting the identifier of the user-provided content into the content object, and inserting the identifier into the outline at that location* (the screenshot of Page 9).

Regarding claims 6, 14 and 22, Vaughn teaches all the claim subject matters as discussed above with respect to claims 4, 12 and 20, Vaughn further discloses *the user-provided content comprises a content entity* (Page 7, the left hand side).

Regarding claims 7, 15 and 23, Vaughn teaches all the claim subject matters as discussed above with respect to claims 4, 12 and 20, Vaughn further discloses *the user-provided content comprises a container* (Page 7, the left hand side).

Regarding claims 8, 16 and 24, Vaughn teaches all the claim subject matters as discussed above with respect to claims 5, 13 and 21, Vaughn reference further comprising the steps of *providing a user interface communicating with the data repository, and providing mechanisms for receiving the user-provided content and specification of a desired location through the user interface* (Pages 9 and 10, How the ezWriter File Works).

Regarding claims 25-30, Vaughn teaches all the claimed subject matters as discussed in claims 1, 24, 9, 12, 17 and 20, Vaughn further discloses *the received user-provided content is not part of the content object* (Screenshot of Page 15) *and wherein the plurality of content entities define the content object as a compilation of related content* (Screenshot of Page 9).

Regarding claim 31, Vaughn teaches *a computer-implemented method for adding user-provided content to a custom content object stored as a plurality of content entities in a data repository* (Page 2 and Page 15) comprising the steps of:

*defining the custom content object by a list of content entity identifiers* (As taught by Vaughn (Page 7), to create a collection of RTF files represented by an ezWriter file that is an index, a user selects [File, New], types in the name that the user would like to call, e.g., “The Trifanton Castle Trilogy” then selects <OK> (FIG. 1). An ezWriter file, and “The Trifanton Castle Trilogy” file are created (FIG. 2). The outline, which is an index file, has only one entry and is displayed in the Outline Window (FIG. 3). Both files were placed into a directory with the same name under the ezWriter directory (FIG. 4). As seen, a collection of RTF files under “The Trifanton Castle Trilogy” file, as *the custom content object is defined by an outline as a list of content entity identifiers*);

*receiving user-provided content, assigning it an identifier, and storing it with its identifier in the one or more object servers* (As taught by Vaughn, ezWriter works in Windows 95, Windows 98, or Windows NT (Page 3, “What is ezWriter”). A user can create an RTF file using Microsoft Word 97, save it and send it to some one. RTF documents can include texts and pictures (Page 6, “Welcome to ezWriter”). Thus, an RTF document contains user-provided texts, e.g., “This is the content of the First Chapter”, is equivalent to the claimed limitation *user-provided content*, which is received by the system from the user that has ezWriter, must be *assigned* a file name, e.g., “First

Chapter", as *an identifier* when saving to "The Trifanton Castle Trilogy" directory. By saving, *user-provided content* (RTF file) with *its identifier* (file name) are *stored in the object server*, e.g., the computer that is used to implement the ezWriter); and

*adding the identifier of the user-provided content to the list, whereby the user-provided content is added to the custom content object* (As further disclosed by Vaughn (Page 8, "How to Add Entries to the Outline"), the file name as *identifier of the user-provided content* could be *added to the outline as the list of content entity identifiers* by selecting the [Outline, Edit] of the Outline Window pull-down menu to view the .ezw file. The .ezw file is opened in Windows Notepad for adding the desired entry, e.g., "First Chapter" (FIG. 8). After saving the file and closing Notepad, by refreshing the outline [Outline, Refresh], the Outline Window displays the added file name (FIG. 9). By selecting "First Chapter" in the Outline Window, *the user-provided content* (RTF file contains texts) is added to "The Trifanton Castle Trilogy" and displayed in the Writer Window (FIG. 10). In short, the technique as discussed indicates the claimed limitation *adding the identifier of the user-provided content to the list, whereby the user-provided content is added to the custom content object*);

*storing said custom content object in said one or more object servers* ("The Trifanton Castle Trilogy" file as *the custom content object is stored in the object server*, e.g., the computer that is used to implement the ezWriter); and

*storing information specifying the custom content object and attribute information concerning the custom content object in the library server* (Page 8, ezWriter.exe as *information specifying the custom content object* and ezWriter.ini as *attribute information concerning the custom content object* are stored in "Program File \ezWriter" as *library server*).

Regarding claim 31, Vaughn teaches *a computer-implemented method for adding user-provided content to a custom content object stored as a plurality of content entities in a digital library having a library server, and one or more object servers* (*user-provided content*, e.g., the content as shown on the right

hand side of Page 2, is added to *content object stored as a plurality of content entities in a data repository*, e.g., “Welcome to ezWriter” as shown in the screenshot of Page 15), the method comprising the steps of:

*defining the custom content object by a list of content entity identifiers* (Page 7, The ezWriter File, paragraph 3, and the left hand side of Page 7, “Welcome to ezWriter” is considered as *custom content object*);

*adding the identifier of the user-provided content to the list* (Page 8, How to Add Entries to the Outline), *whereby the user-provided content is added to the content object* (Page 8, each added identifier will have an RTF file created for it, e.g., when adding “Introduction” to the list, an RTF file with file name “Introduction” as *user-provided content* is created and added to “Welcome to ezWriter”);

*receiving user-provided content* (Page 8, RTF file is received by the system when an entry is added to the outline, e.g., “Introduction” file), *assigning it an identifier* (Page 8, “Introduction” is assigned to the RTF file), *and storing it with its identifier in the one or more object servers* (Page 8 and Page 15, the RTF file and its file name are stored in the computer that implemented ezWriter as *the object server*);

*storing said custom content object in said one or more object servers* (“Welcome to ezWriter” as *the custom content object* is stored in the object server, e.g., the computer that is used to implement the ezWriter); and

*storing information specifying the custom content object and attribute information concerning the custom content object in the library server* (Page 8, ezWriter.exe as *information specifying the custom content object* and ezWriter.ini as *attribute information concerning the custom content object* are stored in “Program File \ezWriter” as *library server*).

The difference between Vaughn and the claimed invention is the order of the process. In Vaughn teaching, the step of adding the identifier is processed before the step of receiving user-provided content, assigning an identifier to the content and storing the content and identifier.

However, as taught by Vaughn, ezWriter works in Windows 95, Windows 98, or Windows NT (Page 3, "What is ezWriter"). A user can create an RTF file using Microsoft Word 97, save it, and send it to some one. RTF documents can include texts and pictures (Page 6, "Welcome to ezWriter"). Thus, instead of creating an RTF file via ezWriter, the user can create an RTF file using Microsoft Word 97. When creating, the RTF files as *user-provided content* is received by the system, the file will be assigned by the user a file name as *an identifier*, and stored in the same directory of ezWriter File in C drive as *the data repository*. After creating the RTF file, the file name will be added to the outline as discussed above as disclosed by Vaughn at Page 9. When the user select an entry in the outline, the corresponding RTF file has file name matches with the entry will be displayed in the Writer Window (theses steps were tested by examiner, and ezWriter gave no error).

Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the Vaughn technique by using a conventional word processor to create an RTF file instead of using ezWriter, and by doing this, use-provided content could be created by a user at one site and transferring to another site that has ezWriter for compiling.

Regarding claim 32, Vaughn teaches all the claim subject matters as discussed above with respect to claim 1, Vaughn further discloses *a plurality of object servers, wherein components of the custom content object are stored in more than one of the plurality of object servers* (As taught by Vaughn, an RTF file can be created using Microsoft Word 97 and send it to some one (Page 6). Thus, by sending the RTF file, the teaching implies that there are at least two computers are used for

delivering information, and these computers are considered as *a plurality of object servers*. And by delivering the RTF files over the network, an RTF file that is a component of an ezWriter file could be created from one computer and transferring to another computer that implemented the ezWriter. In different words, *components of the custom content object are stored in more than one of the plurality of object servers*).

### **Conclusion**

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to HUNG Q. PHAM whose telephone number is 571-272-4040. The examiner can normally be reached on Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, TIM T. VO can be reached on 571-272-3642. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2168

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.



HUNG Q PHAM  
Examiner  
Art Unit 2168

June 2, 2006



Tim Vo

**TIM VO**  
**PRIMARY EXAMINER**